

**CLAIMS**

1. **(Previously Presented)** A job scheduling device for scheduling jobs to run on at least one node of at least one computing platform, comprising:

an enterprise scheduling agent installed on each node and configured to launch execution of jobs submitted to the enterprise scheduling agent;

a presentation system configured to accept and validate parameters identifying at least one job to be submitted for execution on at least one of said nodes; and

a job scheduler configured to allocate at least one job based on said parameters to at least one enterprise scheduling agent and to submit the allocated jobs to said at least one enterprise scheduling agent.

2. **(Original)** The job scheduling device according to Claim 1, further comprising:

a job data management device configured to maintain job data and job histories of jobs submitted to each enterprise scheduling agent.

3. **(Original)** The job scheduling device according to Claim 2, wherein said job histories include information received from each enterprise scheduling agent regarding status of the jobs submitted.

4. **(Previously Presented)** The job scheduling device according to Claim 2, wherein said job data management device is utilized by said job scheduler to set parameters in jobs to be submitted to each enterprise scheduling agent.

5. **(Previously Presented)** The job scheduling device according to Claim 1, further comprising:

a job history repository that saves both jobs and job histories of jobs submitted to each enterprise scheduling agent;

wherein each enterprise scheduling agent comprises,

an agent communicator configured to send and receive messages between said job scheduler and the enterprise scheduling agent,

a job manager configured to setup, launch, run, and manage jobs submitted to the enterprise scheduling agent,

a data manager configured to update and delete data from said job history repository, and

a low level API configured to handle internal functions of said enterprise scheduling agent (LES Agent), file management, and message handling functions.

6. **(Previously Presented)** The job scheduling device according to Claim 5, further comprising:

an enterprise communicator configured to construct and communicate messages between said job scheduler and each enterprise scheduling agent; and

a job data management device configured to maintain job histories of jobs submitted to each enterprise scheduling agent;

wherein said data manager updates said job history via enterprise communicator messages sent from each enterprise scheduling agent to said job data management device.

7. **(Previously Presented)** The job scheduling device according to Claim 1, further comprising:

a command line device configured to accept commands regarding administration of jobs submitted to each enterprise scheduling agent and

a job administration device configured to communicate said command line to at least one of said enterprise scheduling agents for execution.

8. **(Original)** The job scheduling device according to Claim 7, wherein:  
said commands accepted by said command line device include at least one of delete a job and all runs of the job, cancel a job's run, list all jobs by at least one of product code, status, and node, and rerun a job immediately.
9. **(Previously Presented)** The job scheduling device according to Claim 8, wherein:  
said commands accepted by said command line device include context variables; and  
each enterprise scheduling agent converts said context variables according to a current job and job parameters, and executes said commands.
10. **(Previously Presented)** The job scheduling device according to Claim 1, further comprising:  
a point product device configured to provide a communication link between each enterprise scheduling agent and at least one product submitting jobs to said job scheduling device;  
wherein said point product device communicates job status, job logfile, setup, cancel, job parameter functions, and requests between each enterprise scheduling agent and said at least one product.
11. **(Original)** The job scheduling device according to Claim 10, further comprising:  
a job administration device configured to accept command line inputs and communicate said command line inputs to at least one enterprise scheduling agent;  
a job data management device configured to maintain job histories of jobs submitted to each enterprise scheduling agent; and  
an enterprise communicator configured to send messages between at least one of said job scheduler, point product device, job administration device, and job data management device and each of said enterprise scheduling agents.

12. **(Original)** The job scheduling device according to Claim 1, further comprising:

an enterprise communicator configured to send messages between said job scheduler and each of said enterprise scheduling agents.

13. **(Original)** The job scheduling device according to Claim 12, wherein:  
each enterprise scheduling agent is registered at a specific node address that identifies each enterprise scheduling agent with a unique datagroup; and

said enterprise communicator encodes each message with at least one destination corresponding to a datagroup to direct each message to at least one enterprise scheduling agent.

14. **(Original)** The job scheduling device according to Claim 1, further comprising:

a local job repository installed on each of said nodes;

wherein:

each local job repository maintains job and job history information on each job submitted to the node where the local job repository is installed;

each local job repository is updated by the enterprise scheduling agent installed on the node where the local job repository is installed; and

said job information includes job parameters needed to execute each job.

15. **(Original)** The job scheduling device according to Claim 14, further comprising:

a job data management device configured to maintain job histories of jobs submitted to each enterprise scheduling agent; and

a synchronizing device configured to synchronize each local job repository with the job histories maintained by said job data management device.

16. **(Original)** The job scheduling device according to Claim 1, further comprising:

an progress monitor configured to monitor and display execution of at least one of said jobs;

wherein:

said progress monitor provides a visual display of,  
an identification of said job and a current phase of said job,  
a percentage complete of said job, and  
a percentage complete of said current phase.

17. **(Previously Presented)** The job scheduling device according to Claim 1, further comprising:

an auto login device configured to accept login parameters from a user submitting a job;

wherein said login parameters are utilized by each enterprise scheduling agent to launch and execute the job submitted.

18. **(Original)** The job scheduling device according to Claim 1, further comprising:

a notification scripting device configured to execute a notification script having instructions for notifying a user of status or a submitted job;

wherein said notification scripting device includes facilities for creating, editing, and selecting a notification script for a specific job.

19. **(Previously Presented)** The job scheduling device according to Claim 1, wherein:

said presentation system includes,

a GUI interface that accepts user inputs for scheduling and specifying a job to be submitted;

wherein said GUI interface includes facilities for selection and creation of a scheduling calendar, selection of a start date and time, selection of recurring job run intervals, and selection of an immediate job run.

20. **(Original)** The job scheduling device according to Claim 1, further comprising:

a resource management device configured to enable a user to locate and view jobs and job runs.

21. **(Previously Presented)** The job scheduling device according to Claim 20, wherein:

said resource management device includes a GUI for defining an object representing a job, having,

a general properties page having input fields for a label identifying the job, and a description of the job,

a description properties page having a selection field for identifying an icon for representing the job, and

a repository page having a selection field for identifying a time zone for display of job times.

22. **(Original)** The job scheduling device according to Claim 21, wherein:

objects defined by said resource management device comprise,

a hierarchy of folders including at least one of an all jobs folder, a jobs by group folder, a jobs by node folder, a jobs by product folder, a jobs by type folder, and a jobs by user folder.

23. **(Original)** The job scheduling device according to Claim 22, wherein said all jobs folder includes folders, including,

an all jobs any status folder listing jobs regardless of status and associated job history of each job,

an all runs by status folder listing jobs according to status, including completed runs, failed runs, not started runs, preempted runs, running runs, and stopped runs,

a held jobs folder listing jobs that are held and can be scheduled for a later time, and

a scheduled jobs folder listing jobs that are scheduled to run.

24. **(Previously Presented)** The job scheduling device according to Claim 1, wherein:

said presentation system includes, a strategy scheduling window configured to allow a user to view, create, modify, and delete schedules for a strategy.

25. **(Currently Amended)** A method of scheduling jobs across multiple networked computing platforms, comprising:

determining, at a first location, at least one job to be scheduled based on job parameters for the at least one job ~~to be scheduled~~;

sending ~~said~~ the at least one job to at least one enterprise scheduling agent maintained on a selected node of said the computer platforms; and

executing ~~each~~ the at least one job on the selected node under management of ~~said~~ the enterprise scheduling agent; agent;

wherein the first location is communicatively coupled to the selected node by a network.

26. **(Previously Presented)** The method according to Claim 25, further comprising:

monitoring progress of each job executing on the selected node; and  
displaying said progress on a progress monitor.

27. **(Previously Presented)** The method according to Claim 25, further comprising recording each job and a history of each job in a job history repository.

28. **(Previously Presented)** The method according to Claim 27, further comprising:

utilizing a job data management device for,  
retrieving status messages regarding each job sent from each enterprise scheduling agent of a selected node of said job, and  
updating said job history repository based on said status messages.

29. **(Previously Presented)** The method according to Claim 28, further comprising:

maintaining a local job repositories, respectively on each of said nodes, each containing job and job history information for each job submitted to the respective node.

30. **(Previously Presented)** The method according to Claim 29, further comprising:

synchronizing said job history repository with each local job repository.

31. **(Previously Presented)** The method according to Claim 25, wherein said step of determining comprises:

retrieving said job parameters from one of a product and a user interface that collects said job parameters;

validating said job parameters; and

allocating a job based on said job parameters.

32. **(Previously Presented)** The method according to Claim 25, wherein said step of sending comprises:

packaging said job parameters in a communication format; and

transmitting the packaged job parameters from a computing platform where said job parameters are determined to said enterprise scheduling agent maintained on the selected node.

33. **(Previously Presented)** The method according to Claim 25, wherein said step of executing comprises:

setting up the selected node to run an application program identified by said job parameters;

executing said application program on the selected node; and

monitoring progress of said application being executed.



34. **(Previously Presented)** The method according to Claim 25, further comprising:

accepting a command line for administration of a job submitted to at least one of said enterprise scheduling agents; and

communicating said command line to at least one of said enterprise scheduling agents for execution.

35. **(Previously Presented)** The method according to Claim 34, further comprising:

substituting context variables in said command line with data based on said context variable and the job to be administered; and

executing the command line.

36. **(Previously Presented)** The method according to Claim 25, further comprising:

communicating data, including at least one of job status, job logfile, setup, cancel, job parameter functions, and requests for said data between a product and each enterprise scheduling agent.

37. **(Previously Presented)** The method according to Claim 25, further comprising:

registering each enterprise scheduling agent at a node address that identifies the registered enterprise scheduling agent with a unique datagroup;

communicating jobs and job administration commands and requests with each enterprise scheduling agent via messages; and

encoding each message sent to a recipient enterprise scheduling agent with at least one destination corresponding to a datagroup that directs said message to the recipient enterprise scheduling agent.

38. **(Previously Presented)** The method according to Claim 25, further comprising:

retrieving auto login parameters from a user scheduling an auto login job; and  
launching execution of said job utilizing said auto login parameters.

39. **(Previously Presented)** The method according to Claim 38, further comprising:

retrieving a notification script for a job being submitted; and  
executing the notification script on at least one of completion of said job and at a requested status point.

40. **(Previously Presented)** The method according to Claim 25, further comprising:

accepting a scheduling calendar identifying at least one of an execution time and an interval for at least one of said jobs; and  
executing said jobs on selected nodes at the time and interval identified in the calendar.

41. **(Previously Presented)** The method according to Claim 25, further comprising:

providing a description of at least one of said jobs, including a written description, a label, and an icon selected to represent said job; and  
identifying a time zone for display of job times.

42. **(Previously Presented)** The method according to Claim 25, further comprising:

placing information about job times and status in an object containing folders, each folder identifying a categorization of jobs contained therein, including, an all jobs folder, a jobs by group folder, a jobs by node folder, a jobs by product folder, a jobs by type folder, and a jobs by user folder.

43. **(Previously Presented)** The method according to Claim 42, further comprising:

organizing said all jobs folder to maintain additional folders, including, at least one of, an all jobs any status folder listing jobs regardless of status and associated job history of each job,

an all runs by status folder listing jobs according to status, including completed runs, failed runs, not started runs, preempted runs, running runs, and stopped runs,

a held jobs folder listing jobs that are held and can be scheduled for a later time, and

a scheduled jobs folder listing jobs that are scheduled to run.

44. **(Previously Presented)** The method according to Claim 25, further comprising providing a strategy scheduling window that allows a user to view, create, modify, and delete schedules for a strategy.

45. **(Currently Amended)** ~~A computer readable media, having instructions stored thereon that, when loaded into a computer, cause the computer to perform~~ Software for use in scheduling jobs across multiple networked computer platforms, the software embodied in computer readable media that, when executed using one or more computers, is operable to:

~~determining~~ determine, at a first location, at least one job to be scheduled based on job parameters for the at least one job ~~to be scheduled~~;

~~sending said~~ send the at least one job to at least one enterprise scheduling agent maintained on a selected ~~nodes of said~~ node of the computer platforms; and

~~executing each~~ execute the at least one job on the selected node under management of ~~said the~~ enterprise scheduling agent; agent;

wherein the first location is communicatively coupled to the selected node by a network.

46. **(Previously Presented)** A job scheduling device for scheduling jobs to run on at least one node of at least one computing platform, comprising:

enterprise scheduling means installed on each node and configured to launch execution of jobs submitted to the enterprise scheduling means;

presentation means configured to accept and validate parameters identifying at least one job to be submitted for execution on at least one of said nodes; and

job scheduler means configured to allocate at least one job based on said parameters to at least one enterprise scheduling means and to submit the allocated jobs to said at least one enterprise scheduling means.

47. **(Previously Presented)** The job scheduling device as recited in Claim 1, wherein the presentation system comprises a Graphic User Interface Application Program Interface (API GUI).

48. **(Previously Presented)** The job scheduling device as recited in Claim 46, wherein the presentation means comprises a Graphic User Interface Application Program Interface (API GUI).

49. **(Previously Presented)** A job scheduling system for scheduling jobs to run on at least one node of at least one computing platform, comprising:

an enterprise scheduling agent installed on each node and configured to launch execution of jobs submitted to the enterprise scheduling agent;

a presentation system configured to accept and validate parameters identifying at least one job to be submitted for execution on at least one of said nodes; and

a job scheduler configured to allocate at least one job based on said parameters to at least one enterprise scheduling agent and to submit the at least one allocated job to said at least one enterprise scheduling agent so that the at least one allocated job can be executed on a node having the at least one enterprise scheduling agent to which the job was allocated and submitted.

50. **(Currently Amended)** A job scheduling method for scheduling jobs to run on at least one node of at least one computing platform, comprising:

installing an enterprise scheduling agent on each node for launching execution of jobs submitted to the enterprise scheduling agent;

at a first location:

accepting and validating parameters identifying at least one job to be submitted for execution on at least one of said the nodes; and

allocating at least one job based on said parameters to at least one enterprise scheduling agent installed on a selected node;

submitting the at least one allocated job to said the at least one enterprise scheduling agent; and

executing the at least one allocated job on ~~a node~~ the selected node having the at least one enterprise scheduling agent to which the job was allocated and ~~submitted.~~ submitted;

wherein the first location is communicatively coupled to the selected node by a network.